

rejection, as required by the Federal Circuit and the Supreme Court.¹

Maeshima et al.

U.S. Patent No. 6,092,113 to Maeshima et al. fails to disclose or suggest the claimed “controlling supply of data packets to the cryptographic module” for *generation* of the encrypted packets, as argued in the rejection. To the contrary, Maeshima et al. teaches away from the claimed controlling supply of data packets to the cryptographic module by assuming the packets received and queued by any of the routers 300 (described with respect to column 4, lines 1-18 and column 5, lines 45-64) have already been encapsulated within a tunnel by a tunnel endpoint.

In particular, Maeshima et al. consistently teaches that construction of the IP tunnel between tunnel endpoints is performed before any resource reservation can be completed.² Maeshima et al. specifies with respect to Figure 1 that “[e]ach application 202 on both LAN 200A and 200B is encapsulated at the start point of the IP tunnel” (column 3, lines 17-19), and that “[t]he IP tunnel 101 is set by adding an IP tunnel function only on a machine (IP tunnel server) at both ends of the IP tunnel 101” (col. 4, lines 44-46). Moreover, column 4, lines 50-59 illustrate that the IP tunnel server can be implemented as distinct servers 203, illustrated in Figure 9(b), that establish the tunnel 101 before any of the routers 300 receive and queue the encapsulated packets. Any encryption would therefore be performed by the tunnel endpoints, and therefore before any of the routers 300 queue the encapsulated packets (see, e.g. col. 1, lines 25-29; col. 4, lines 44-65).

¹ “[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.” *KSR Int’l v. Teleflex, Inc.* No. 04-1350, Slip. op. at 14, 82 USPQ2d 1385, 1396 (U.S. Apr. 30, 2007) (*quoting In re Kahn*, 78 USPQ2d 1329, 1336 (Fed. Cir. 2006))

² See, e.g., Abstract at lines 1-3; col. 3, lines 1-13; col. 5, lines 28-42 (especially lines 37-39); col. 7, lines 49-53; col. 7, line 66 to col. 8, line 7; col. 8, lines 19-28 and 41-49; col. 9, line 14 to col. 10, line 2.

Maeshima et al. also consistently teaches that the packets are received and queued by the routers 300 after encapsulation by the tunnel endpoints. Maeshima et al. teaches (e.g., with respect to Figs. 1-4, Figs. 9(a), 9(b) and Fig. 10) that each of the routers are "on the IP tunnel 101" and support Reservation Resource Protocol (RSVP) (see, e.g., column 3, lines 14-16). Figure 2 also describes that the processor 307 within a router 300 receives encapsulated packets 311 that already have the original IP datagram 309 encapsulated with a tunnel header 310 identifying the IP tunnel endpoints (see, e.g., column 5, lines 23-27). Hence, the packets 311 that arrive at each input interface of a router and that are allocated by the processor 307 to an input buffer 301 or 302, as illustrated in Figs. 2-4 and described at column 5, lines 45-64, have already been encapsulated by the tunnel endpoint. Consequently, even if a router (e.g., 300A of Fig. 1 or 9(a)) performs as a tunnel endpoint (see col. 3, lines 1-10), the foregoing demonstrates that the encapsulation and encryption by the tunnel endpoint is performed before the processor 307 of Fig. 2 receives the encapsulated packet 311.

Hence, Maeshima et al. teaches away from the claimed feature of controlling supply of data packets *to the cryptographic module* based on assigning, for each secure connection, a corresponding queuing module, and outputting *to the cryptographic module* the group of data packets from each corresponding queuing module. Maeshima et al. teaches away from this claimed feature because Maeshima et al. consistently teaches that the packets are received and queued in the buffers (e.g., 301) after encapsulation by the tunnel endpoints.

For this reason alone the obviousness rejection must be withdrawn because Maeshima et al. fails to disclose or suggest the claim features as argued in the rejection.

As admitted in the rejection, Maeshima et al. fails to teach that each encrypted packet successively output from the cryptographic module has a corresponding successively-unique sequence number; or the claimed feature of reordering, in each queuing module, a corresponding group of the data packets associated with the corresponding secure connection according to a determined quality of service policy and based on a corresponding assigned maximum output bandwidth for the corresponding queuing module. In fact, this admission demonstrates that Maeshima et al. teaches away from the claimed feature because one skilled in the art would

recognize that the packets following encryption by the tunnel endpoints cannot be reordered, else the reordered packets would be dropped by the receiving cryptographic module.

MeLampy et al

Applicant traverses the assertion that the MeLampy et al. teaches the claimed feature that each encrypted packet successively output from the cryptographic module has a corresponding successively-unique sequence number. To the contrary, paragraph 15 of MeLampy et al. teaches away from this claimed feature by specifying "assigning a sequence number to a first multi-media data flow packet received by a first endpoint, ... pseudo-randomly *shuffling* the sequence number of the first data flow packet; and transmitting the pseudo-randomly shuffled sequence number to a second endpoint." This summary at paragraph 15 is described in further detail with respect to step 302 of Figure 4 and paragraphs 40-55.

In particular, paragraph 40 explicitly specifies that the sequence numbers are within the RTP flow packets that are generated by the phone 114 (see para. 31) and received by the media router 118 of Figure 3:

[0040] As shown by block 302, sequence numbers within the RTP flow are randomly shuffled. In accordance with the preferred embodiment of the invention, a sequence number is assigned to each RTP multi-media data flow packet within an RTP flow such that when an RTP multi-media data flow packet is received, the associated sequence number may be determined. Randomization code is utilized to provide random shuffling of the sequence numbers. Preferably, the random shuffling is algorithmically predictable if a key to the randomization code is known. Therefore, since the randomly shuffled sequence numbers are algorithmically predictable if the key is known, the sequence numbers really are not randomly shuffled but are instead, pseudo-randomly shuffled.

Moreover, para. 55 of MeLampy et al. explicitly teaches that the sequence numbers within the RTP flow received by the media router 118 are encrypted by replacing the original sequence number with encrypted sequence numbers having an example sequence (illustrated in paragraphs 43-53):

[0055] Applying this mapping to the step of randomly shuffling sequence numbers within an RTP multi-media data flow (block 302), the first RTP multi-media data flow packet

has a sequence number 1888747329 (which maps to 1), the second packet has a sequence number 1601588182 (which maps to 2), and so on. Using this algorithm, the receiving side may produce a sequence of expected sequence numbers and restore them. As an example, a sender that is transmitting an original sequence number of 1 (or a salt value of 1) may replace the original sequence number with an encrypted sequence number of 1888747329. The encrypted sequence number of 1888747329 may then be transmitted to a receiving side.

(Para. 55, lines 1-13).

As apparent from paragraphs 43-53, the encoded sequence numbers are not successively-unique sequence numbers (e.g., 1, 2, 3, etc.), as claimed; to the contrary, the disclosed sequence numbers in para. 43-53 are pseudorandom numbers (e.g., 1888747329, 1601588182, 1967410106, etc.) that are decoded at the receiver.

Hence, MeLampy et al. fails to disclose or suggest the claimed feature that “each encrypted packet successively output from the cryptographic module [has] a corresponding successively-unique sequence number”. For this reason alone the §103 rejection must be withdrawn because the applied reference fails to disclose the claimed feature as asserted in the rejection.

In fact, MeLampy et al. further teaches away from the claimed feature that “each encrypted packet successively output from the cryptographic module [has] a corresponding successively-unique sequence number” by teaching that packets can be resequenced in step 306 after the shuffling of sequence numbers in step 302 and encryption of data port addresses in step 304: [a]s shown by block 306, resequencing of the multi-media packets is then performed ...” (para. 60, lines 1-2); further, MeLampy explicitly teaches that “in accordance with the re-sequencing of multi-media data packets, the multi-media data packets may be transmitted in any order desired, including, but not limited to, 2, 5, 4, 1, 3, etc.” (Para. 61, lines 7-10).

Hence, the §103 rejection must be withdrawn because MeLampy et al. teaches away from the claimed feature that “each encrypted packet successively output from the cryptographic module [has] a corresponding successively-unique sequence number”.

Applicant further traverses the Examiner’s assertions that MeLampy discloses or suggests

the claimed *reordering* the corresponding group of the data packets (that is *output to the cryptographic module*) according to a determined quality of service policy and the corresponding assigned maximum output bandwidth. Rather, MeLampy describes in para. 33 that the traffic manager 206 is used simply for “measuring and enforcing IP session data flow rates”, where “once a forwarding decision is made, the traffic manager 206 queues the received packet into its respective IP flow and associated priority.”

Paragraph 34 (cited by the Examiner) also specifies that maximum data rates are enforced not by *reordering* the data packets, but by “either dropping packets or marking them as eligible for discarding if they are outside a bandwidth allocated for the data flow.” (Para. 34, lines 5-7). Further, the multi-media router 118 may be instructed by a session router 116 to enforce an allocated bandwidth and bit rate, such that “if data is received at a higher bit rate than allowed by the session router, the data received at the higher bit rate is not transmitted” (para. 34, lines 10-12). For this reason alone the §103 rejection must be withdrawn because the applied reference fails to disclose the claimed feature as asserted in the rejection.

Hence, MeLampy teaches that the encrypted packets can be transmitted in any order desired; further, MeLampy fails to teach or suggest that the packets should be reordered and output *to the cryptographic module* according to the determined quality of service policy and based on the corresponding assigned output bandwidth; rather, MeLampy simply queues the packets for transmission, with no suggestion that the packets should be ordered *prior to encryption*, as claimed.

The Hypothetical Combination

As apparent from the foregoing, the hypothetical combination fails to disclose or suggest the claimed features of controlling supply of data packets to the cryptographic module by assigning, for each secure connection, a corresponding queuing module, reordering in each queuing module the corresponding group of data packets according to a determined quality of service policy, and outputting to the cryptographic module the group of data packets from each corresponding queuing module, where each encrypted packet successively output from the

cryptographic module has a corresponding successively-unique sequence number. To the contrary, the hypothetical combination simply would provide media packets having encrypted sequence numbers that can be resequenced "in any order desired" (paragraph 61 of McLampy et al.) prior to being queued within RSVP buffers in a router (as described in Maeshima et al.).

Hence, the rejection has failed to demonstrate that "there was an apparent reason to combine the known elements *in the fashion claimed* by the [claims] at issue [where] this *analysis should be made explicit*." *KSR Int'l v. Teleflex, Inc.* No. 04-1350, 550 U.S. ___, Slip. op. at 14, 82 USPQ2d 1385, 1396 (U.S. Apr. 30, 2007). To the contrary, the rejection simply presents a hypothetical combination that teaches no more than "the predictable use of prior art elements according to their established functions," *Id.*, with no disclosure or suggestion of the claimed features as a whole.

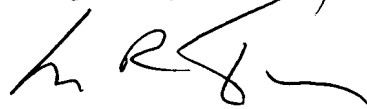
For these and other reasons, the §103 rejection should be withdrawn.

It is believed the dependent claims are allowable in view of the foregoing.

In view of the above, it is believed this application is in condition for allowance, and such a Notice is respectfully solicited.

To the extent necessary, Applicant petitions for an extension of time under 37 C.F.R. 1.136. Please charge any shortage in fees due in connection with the filing of this paper, including any missing or insufficient fees under 37 C.F.R. 1.17(a), to Deposit Account No. 50-1130, under Order No. 10-008, and please credit any excess fees to such deposit account.

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'L. R. Turkevich', with a stylized flourish at the end.

Leon R. Turkevich
Registration No. 34,035

Customer No. 23164
(202) 261-1059
Date: December 4, 2007

Response filed December 4, 2007
Appln. No. 10/759,182
Page 7